Amendments to the Specification:

Please replace the Abstract of the Disclosure with the following amended Abstract of the Disclosure. A new Abstract is annexed hereto on a separate sheet.

Methods and apparatus are provided for automatically loading drivers on computers for devices coupled to a communication medium such as an IEEE 1394 serial bus. Operating systems running on the computers detect when a device is connected to the communication medium. The operating systems then determine whether one of the drivers stored in memory on the device will enable the computers to utilize the device. The operating systems will then load the appropriate drivers for their respective computers and allow access to the device. If none of the drivers on the memory are suitable for a particular computer, then its operating system will prompt a user to manually insert a computer-readable medium containing the appropriate drivers. Thus, the present invention provides methods and apparatus for automatically loading device drivers.

Please replace the paragraph beginning at page 5, line 17, with the following amended paragraph:

An IEEE 1394 interface ~~140~~ 142 may also be provided. The IEEE 1394 interface ~~140~~ 142 couples an IEEE 1394-compliant serial bus 145 to the system bus 130 or similar communication bus. The IEEE 1394-compliant serial bus 145, as known in the art, allows multiple devices ~~150~~ 148 to communicate with the computer 100 and each other using high-speed serial channels. The IEEE-1394 high-performance serial bus standard is based largely upon the internationally adopted ISO/IEC 13213 (ANSI/IEEE 1212) CSR Architecture Specification and the IEEE 1394-1995 Serial Bus Specification, the teachings of which are herein incorporated by reference. Additional buses such as the PCI bus can be provided in computer 100 and interfaced to the IEEE 1394 and other buses.

Please replace the paragraph beginning at page 9, line 1, with the following amended paragraph:

A flow chart illustrating an exemplary process of automatically loading device drivers is shown in FIG. 3. An operating system running on a computer 200, 202 is first started at S300. After the operating system is running, the computer 200, 202 can then detect at S302 whenever a

device 210, 212 is connected to the bus 204. The device 210, 212 is available to be enumerated the moment it is plugged in. It is always free in this sense since the invention does not need to be able to determine what drivers to load in the normal case. That is in the normal case step S304, checking whether the device is free will be yes and the abnormal case of waiting for the device to be free in step S306 will be avoided. Thereafter, the operating system will check to see if the CROM 218, 220 stores drivers or pointers to drivers at S308. If the CROM does not store the drivers or pointers, then the operating system will conclude that the peripheral device is a legacy device. Consequently, the operating system will prompt its user to manually load the driver in the standard prior-art way of loading drivers S310.